Collaborative Virtual Environments For Training: A Unified Interaction Model For Real Humans And Virtual Humans

Andrés Saraos Luna, Valérie Gouranton, Bruno Arnaldi

Collaborative Virtual Environments For Training: A Unified Interaction Model For Real Humans And Virtual Humans

Andrés Saraos Luna, Valérie Gouranton, and Bruno Arnaldi

Université Européenne de Bretagne, France INSA, Inria, Irisa, UMR CNRS 6074, F-35708 RENNES andres.saraos_luna@irisa.fr, valerie.gouranton@irisa.fr, bruno.arnaldi@irisa.fr

Abstract. Our work ponders on collaborative interactions in Collaborative Virtual Environments for Training, with an emphasis on collaborative interactions between Real Humans and Virtual Humans working as a team. We put forward a new model of collaborative interaction and a set of tools that describes and defines such interactions.

1 Introduction

The use of virtual reality for training offer many assets: low cost and low risk, no need for physical and often costly equipment, possibility of various degree of teamwork [7]. In Collaborative Virtual Environments for Training (CVET), a group learn and practice the completion of a task as a team using all the assets provided by virtual reality. Depending on the system, the task can be extensively defined in a scenario or be defined as a global goal to pursue. Teammates can work together even though they are not physically in the same place using network, the task can be repeated as many times as necessary. The collaborative actions between teammates range from planning their next actions together to collaboratively manipulating objects, to remove a heavy piece of machinery for instance.

Moreover, teammates can either be human beings or computer generated agents. Teamwork between human beings and autonomous agents has been for some time a subject of interest [6]. Agents can be used in a variety of ways in this context. As collaborators, they share a common goal with their teammates and work towards its achievement. As troublemakers, they will try to interfere with the accomplishment of the task forcing its teammates to find alternate solutions or to deal with the additional issues. As teachers and guides, they can provide contextual help whenever they're needed. The different roles that agents can play in a team can be sorted in two main groups [26]:

- the role of an equal team member.
- the role of assistants to either the team as a whole or an individual team member,

In CVETs all team members are on the same ground, working towards the same goal. CVETs therefore focuses on the first case.

Main Results Our work concentrates on collaborative interaction in CVETs with a focus on collaborative interactions, more precisely collaborative manipulation of objects, between Real Humans (RH) and Virtual Humans (VH) working as a team. We propose a new collaborative interaction model and from it construct a set of tools to describe and define such collaborative interactions.

Organization In section 2 we investigate related works. Throughout section 3 we present the collaborative interaction model constituting our contribution all the way to a demonstrator. Finally, we conclude in section 4.

2 Related Work

An overview on existing CVETS is presented in section 2.1 leading to an analysis on two points: the collaborative interactions offered to RHs and VHs, and the level of teamwork achieved. We then look at how behavior and interactions between objects can be described in section 2.2, and how collaborative interaction in virtual environments has been handled in previous works in section 2.3.

2.1 CVETs

In this section, we focus on CVET applications that emphasize on collaborative training and look at their interaction capability and the way RHs can interact with VHs. Three different collaboration levels in virtual environments have been identified [18]:

1. Awareness of team members' presence in the virtual space and communication between them.
2. Individual interaction of each user with objects in their environment.
3. Collaborative and codependent interaction of various users with the same object.

Scarcely any CVET implements level 3 collaboration. In the COVET [21] application, users learn collaboratively by looking, communicating and thus assisting one of the trainees who is the only one able to interact with the world. This is also the way most medical training applications implement collaboration [4], with one teacher performing the medical operation while students can switch viewpoints and communicate to learn the procedure. One exception is the CyberMed/CybCollaboration framework [25] that authorizes concurrent inputs from various users to the same object, although the only known application of the framework does not seem to use this asset. All these applications share another common point: none of them offer to interact with VH, thus not providing any collaborative teamwork with virtual users.

The Dugdale's fire training simulation [10] stops at the second level of collaboration and was aimed to support collaboration between RH and VH. However, the demonstrator was implemented using only RH and succeeding works focused on agent-only collaborative work for emergency management [12].

The MASCARET model implemented in the SecureVI fire-training application [23], proposes a framework for designing pedagogical software with team training. The model supports interaction through their “interaction network” where a source agent and a target agent exchange information about the interaction. Although this model seems to support collaborative input, it is not explicitly addressed and not implemented in the SecureVI software.

We note that the collaborative work is often viewed as one user interacting with the virtual environment while the other users can share his view, feedback and/or actions. Even though some models state to authorize joint manipulations of objects, it is scarcely addressed and even less so implemented.

2.2 Interaction Description

The research in behavioral animation can be narrowed down to 3 approaches [9], stimulus response systems, rules systems and state machines. In stimulus response systems, each stimulus evolves through a grid of interconnected nodes to generate a behavior from the system, as done with the SCA network [16] to control the behavioral animation of a VH. Although these networks offer responsive and reusable behaviors, the underlying mechanisms produce behavior with low-level of both abstraction and control.

Rules systems approaches like Reynolds' flock of birds [24] or Blumberg's dog modelization [3] propose to assign simple rules to model behavior. Complex group behavior emerges from the rules defined for each entity. This gives us a high level of abstraction to describe entities' behavior but low re-usability as the rules are designed for each system. As the environment becomes more complex and the number of rules increases, run-time is slowed thus making it difficult to use in an interactive context.

Although simple state machines are deemed impractical to describe complex behaviors [17], hierarchical and parallel-state machines like the HPTS language [8] allow a precise tuning of behavior while maintaining re-usability. However the process of designing complex automata can be long and tedious.

The Simulation and Training Object-Relation Model (STORM) [19] is an hybrid method built from the state machine family and aims to regroup the properties of existing methods in one model in order to obtain a generic and reusable solution. STORM is composed of two parts : a behavioral model to easily describe reusable objects, and an interaction engine which defines a standard process that allows objects to interact with potentially unknown objects. The capacities are designed as to be pluggable into any existing STORM object with few parameters to configure. Any object can then be enhanced with existing capacities, allowing the description of behavioral objects in a generic and reusable way.

The main drawback of the existing approaches is that collaborative manipulations are not explicitly taken into account by the existing techniques, which makes the description of collaborative interactions between users difficult.

2.3 Collaborative Interaction

Interaction methods for individual users serve as a ground base for collaborative techniques. We only consider virtual hand and virtual pointers technique as they prove to be more suitable for a collaborative setting [2]. However, these techniques need to be re-factored for collaborative usage.

A first approach is to restrain users to different degrees of freedom to interact with the object [22]. If one user modifies an object's orientation, this parameter becomes unavailable to a second user who would only be able to alter the object's position in space. As such, it is still possible to use single user techniques for users with different points of view in the environment. The main drawback is that it hurts the credibility of the interaction as real world interactions are not cleanly disjoined in this way.

We thus consider interaction techniques allowing concurrent access to the same properties of an object. A variety of techniques exists: by averaging forces applied to objects via multiple virtual hands, virtual 3D cursors or variations of the virtual ray metaphor. One of the main issues of such techniques is the feedback given to the users in return of the actions, both theirs and their teammates'. Without force feedback, it can be difficult to offer the users a proper one.

An interaction as a bidirectional communication between two objects, named tool and interactive object [13][1]. The tool sends commands and parameters to an interactive object that is responsible for the treatment of the commands. Pluggable extensions to objects convey the ability to interact or be interacted upon, with no prior knowledge of the available parameters needed to initiate the control. When various tools control the same parameters of a single interactive object, the latter deals with concurrent inputs as he sees fit and the sends feedback of the final result to the connected tools. The extension mechanism makes it generic enough to be adapted on existing models although the presented implementation depends heavily on the OpenMask [1] platform.

We raise one main common issue amongst all the presented works: none of them seems to consider VHs as users of the interaction methods. It is always assumed that a human being is the source of interaction, and as such the methods may need some tuning to allow virtual users.

2.4 Synthesis

To conclude this state of the art, let us underline the key points of our analysis:

1. Interactions between objects in the virtual world can be described in a generic way, but the available models don't take *collaborative manipulation* into account.
2. Objects of the environment can be controlled by human beings, single-handedly or collaboratively. Virtual humans are not considered as potential *users* by the exposed methods of interaction.

[1] COllaboRative Virtual Environment for Technical Training and Experiment, http://corvette.irisa.fr/

3. *Collaboration between Real Humans and Virtual Humans* is scarcely addressed in existing CVETs, and even less so when considering collaborative manipulation of an object by users of different nature.

3 Unified Interaction Model for Real Humans and Virtual Humans

After a quick overview of our objectives (section 3.1) and a presentation of our experimental use case (section 3.2), we define a new collaborative interaction model, named **Collaborative-STORM** (C-STORM), based on the STORM model (section 3.3). C-STORM collaborative objects can be manipulated from multiple sources, thus achieving a level 3 collaborative interaction. We then use this model as a ground base to define the **Interaction Mediator** (InterMed) (section 3.4), an entity connecting the user, real or virtual, to the virtual environment. The InterMed is an inhabitant of the virtual world and as such is able to perceive and interact with its environment. We finally expose an implementation of these concepts in a demonstrator (section 3.5).

3.1 Overview

The construction of our novel collaborative manipulation model aims for the following objectives:

1. *The construction* of a model capable of describing generic collaborative interactions and behaviors.
2. *The unification* of Virtual Humans and Real Humans as interactors and users of the CVET.
3. *The implementation* of those collaborative methods in a CVET.

Connecting RHs and VHs to the virtual world through a mediator offers a number of benefits. From a software engineering point of view, it greatly simplifies the design process. All the actions are performed by the InterMed, regardless of the nature of the commanding user and thus only need to be defined as such. From a behavioral point of view, VHs interacting through the C-STORM model make no difference between RHs and VHs as everyone is embodied by an InterMed. Interactions between users in the virtual world are consequently abstract from the nature of the user. From a usage point of view, RH using the system can collaborate with the other teammates in the virtual world while being unaware of their real nature without impeding their task.

3.2 Experimental Use Case

Throughout our contribution section, we provide examples extracted from our experimental use case. It provides a collaborative setting in which various people have to build a small piece of furniture. Some steps can be carried out by one person and some others need two or more people to collaboratively interact. Some knowledge concerning the handling of various tools (screwdriver, hammer...) is needed. The key step in which we will focus for our examples is the screwing process, where one user needs to take control of a screwdriver in order to screw some pieces of furniture together.

3.3 The Collaborative-STORM Model

The **Collaborative-STORM** model takes advantage of the perks of the STORM model and constructs upon it to address its lack of description of collaborative manipulation of one object by various users. C-STORM displays new capacities that will give C-STORM objects the ability to control other objects, and to share this control with multiple other sources. This mechanism allows users, who are considered as objects, to share the control of an object, thus achieving *collaborative manipulation*. A small example of how STORM connects objects through relations is provided in Fig1.

Two new capacities are introduced by the C-STORM model: the **Interactor** capacity and the **Interactive Object** capacity. The Interactor capacity grants C-STORM objects the ability to take control of objects disclosing the Interactive Object capacity. The latter offers a list of controllable parameters and treats the commands it receives. This mechanism grants users the aptitude to take control of objects in the virtual environment. From now on, we will refer to objects enhanced with the according capacity as *interactors* and *interactive objects*.

The control mechanism is encapsulated in a dedicated relation. The **Collaborative Interaction Protocol Relation** (CIPR) connects Interactor and Interactive Object capacities, as illustrated in Fig. 2. The CIPR opens a communication channel between interactors and interactive objects. Through this channel, they play out a protocol that determines the terms of the control contract between them. Once the connection is established, the **Control Relation** (CR) takes over. CR is a sub-relation contained in the CIPR. It delivers all parameters, commands and feedback sent from one object to the other, as long as no ending signal is received from within the CIPR.

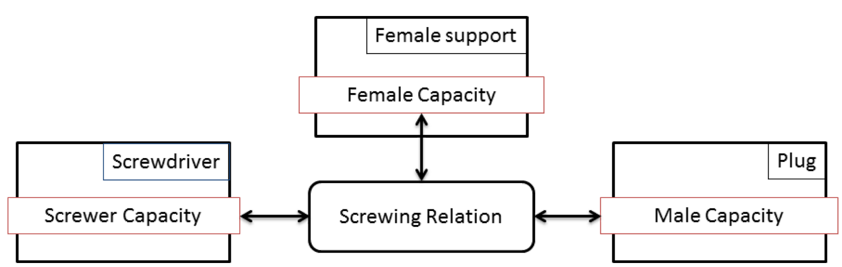

Fig. 1. Example: a STORM Relation: behavioral objects linked together

The control protocol can be split into the following steps:

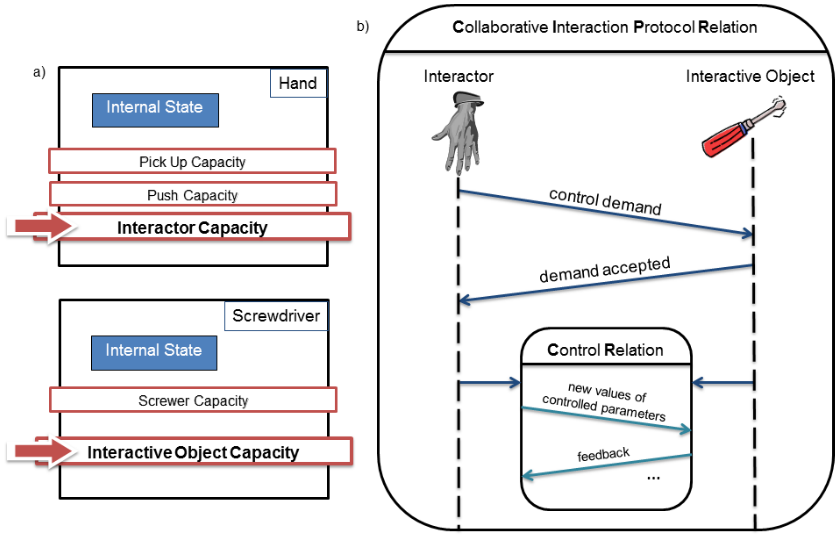

Fig. 2. a) Two C-STORM objects with the Interactor and Interactive Object capacities. b) The CIPR plays out the communication protocol between 2 C-STORM objects, then the CR conveys the different signals and commands from one object to the other

- The interactor initiates contact, asking for a control authorization and the parameters available to him.
- The interactive object considers the demand. If it is accepted, a clearance is sent along the parameters it offers for modification.
- The CIPR inititates the CR, which will serve as a transmission channel for new parameters values.
- When the CIPR receives an ending signal, the control is terminated.

Each interactive object can accept multiple connections from multiple interactors. The way concurrent inputs are evaluated is parameter dependant and is handled internally by the interactive object. In the same way, interactors can be connected to various interactive objects and consequently control several objects simultaneously.

C-STORM objects effectively achieve *collaborative manipulation* through the Interactor and Interactive Object capacities. Collaborative objects can be constructed directly from existing objects thanks to the capacity mechanism, facilitating the design process and offering *reusable collaborative behaviors*.

Example We modelize a sequence extracted from our use case presented is section 3.2, illustrated in Fig.3. To build the piece of furniture, it is required to screw a few pieces together. To do so, we use a Screwing Relation that connects a screwdriver with the Screwer capacity, a female support with a Female capacity and a plug with a Male capacity. The screwdriver and the male plug need to be put in a certain position in order to proceed with the Screwing Relation. Interactor capacities are connected to the screwer tool and the male plug through

CRs. This allows the interactors to modify the position and orientation in space of the objects they control. Once the screwdriver and the male plug are in a certain area of effect defined in the Screwing relation, the behavior associated with the relation can proceed.

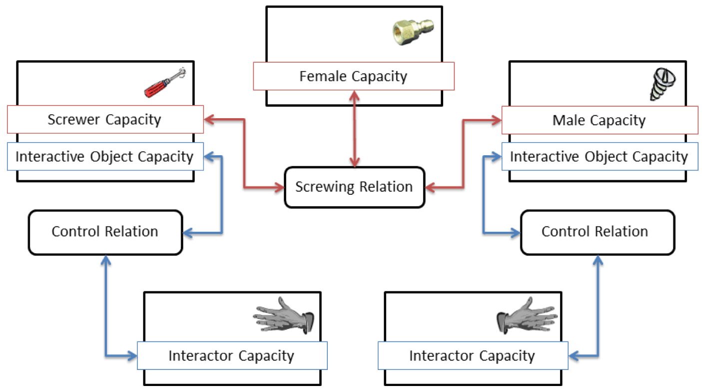

Fig. 3. The screwing relation example, two hands controlling the screwdriver and the screw thanks to Control Relations

3.4 The Interaction Mediator

Using the C-STORM model defined in the previous section, we construct a unified interaction model for both RHs and VHs by introducing a new C-STORM object, the **Interaction Mediator**. This new tool allows us to define a unified model of interaction for RHs and VHs.

Concept We define in the term **users** either real or virtual humans involved in the teamwork. This decision arises from the role VHs play in the team. As they assume the role of an equal team member, we propose that they share the same characteristics. We thus choose to dissociate the users from the interactive actions within the virtual world. Users are considered as living outside the virtual environment; as such, they can't interact directly with objects in the environment and don't have knowledge about the objects either. They are independant of any kind of embodiment in the virtual world.

We thus have on the one hand objects in the virtual environment that can interact with each other, and on the other hand users that cannot alter the environment. We introduce a specific object of the environment, *a mediator* closing the bridge between the users and the virtual world.

This mediator is more than a simple embodiment though. It is an entity living in the virtual space, capable of perceiving the objects surrounding it and storing all the perceived data into its knowledge base. Unlike the users, the mediator can thus perceive the virtual environment and as an object it can interact with other objects. However, being deprived of any kind of decision process it will never initiate interactions on its own.

Users, either real or virtual, can consequently connect to a mediator to interact with the virtual world through the mediator's interaction capabilities and use its knowledge to aid their decision process. Mediators serve as a *gateway* unifying the interaction commands sent by the users, and relaying the knowledge acquired from within the virtual world. We will refer to this mediator as an **Interaction Mediator** (InterMed) as illustrated in Fig.4.

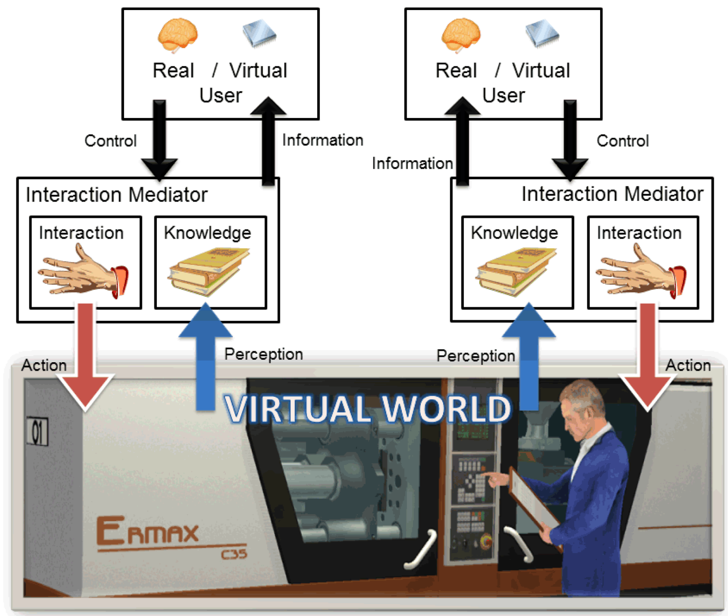

Fig. 4. The InterMed, a mediator between the users and the virtual world

Model Having defined our user notion and its connection to the virtual world, we integrate the concept into our C-STORM model. Our key concept, the InterMed, is constructed as a special C-STORM object. As stated before, users control InterMeds to interact with the virtual world. Translated in our modelization, this means users are represented as interactors that can take control of InterMeds, through the CIPR relation described in section 3.1. The user is embedded in a C-STORM object that transmits his commands to the connected InterMed. As such, it needs a list of the parameters and commands it can convey, encapsulated in a dedicated activity. The data is then transmitted through the Interaction capacity to the controlled InterMed.

The InterMed can perceive and interact with other objects, and receives control inputs from the connected user. The interaction mechanism is assured by the Interactive Object and Interactor capacities to let users control the InterMed and to let the InterMed interact collaboratively with other objects. The perceptive capabilities of the InterMed are encapsulated in a new capacity, the Perceptive capacity. It grants the ability to perceive the environment via various perceptive channels. All the capacities of the InterMed can be adjusted by the designer at will. Any type of InterMed can be designed by fine tuning its capacities, from a floating eye to a full-fledged humanoid. The awareness of the Perceptive Puppet fully depends on the implementation of its Perceptive capacity.

This model follows our concept key points: users interact with the virtual world by controlling InterMeds. The knowledge acquired by the InterMed through its perceptive channel is available for the user that controls it, allowing informed decision making. InterMeds can interact alone or collaboratively thanks to the C-STORM mechanisms described in section 3.3. Consequently, as InterMeds can be controlled by either a RH or a VH, collaborative manipulation occurs seamlessly even with users of different nature.

3.5 Validation: A Proof of Concept

We demonstrate our InterMed concept by implementing the model described in section 3.3 in our experimental platform. This platform is developped in the context of the CORVETTE [1] project, a Research collaboration. The project is a follow-up of the GVT [2] project, a VR platform for building virtual training environments, that aims for collaborative training with both real and virtual humans [20][14][15]. The CORVETTE project has been presented during the LAVAL Virtual 2012 event [3].

The demonstrator serves as an early implementation of our collaborative tools in an existing CVETs, a first step to full-fledged collaboration between RHs and VHs in CVETs and this paper does not aim at a full explanation of our validation. Our demonstrator (Fig.5) allows for a VH and RH to collaboratively manipulate a piece of furniture. Each user is embodied by an InterMed.

4 Conclusion

The analysis of existing work highlighted various gaps in collaborative training between Real Humans and Virtual Humans in Collaborative Virtual Environments for Training. A new model of collaborative interaction was needed in order to easily describe and integrate collaborative actions between RHs and VHs in CVETs. Throughout this paper, we presented:

1. **Collaborative-STORM**, a generic collaborative model of interaction between objects.

[2] Generic Virtual Training, https://www.gvt-nexter.fr/
[3] http://www.laval-virtual.org/

Fig. 5. Left: Our demonstrator in action, with two humanoid InterMeds collaboratively manipulating a piece of furniture. Right: The CORVETTE project as exposed in the LAVAL Virtual 2012 event.

2. **The Interaction Mediator**, a specific C-STORM object serving as a gateway between users either real or virtual and the virtual environment.
3. **A demonstrator** showcasing an implementation of our concepts in an existing CVET platform, the GVT project.

These contributions successfully address the identified needs. Moreover, the work hereby presented opens doors to many upgrades: pushing forward with our unified model to propose an effective interchangeability between users during teamwork, proposing various profiles of InterMeds by altering or improving their capacities or even considering mutual control of an InterMed by various users. The presented work and its subsequent follow ups could also be considered in a gaming context, although it is not the focus of this paper.

Acknowledgments This work was supported by the French Research National Agency project named CORVETTE (ANR-10-CONTINT-CORD-012).

References

1. Aguerreche L., Duval T., Arnaldi B.: A description of a dialog to enable interaction between interaction tools and 3D objects in collaborative virtual environments VRIC 2009 (2009) 63–73
2. Aguerreche L., Duval T., Lécuyer A.: Comparison of Three Interactive Techniques for Collaborative Manipulation of Objects in Virtual Reality CGI 2010 Computer Graphics International (2010)
3. Blumberg B., Todd P., Maes P.: No bad dogs: Ethological lessons for learning in hamsterdam (1996), 295–304
4. Boulanger P., Wu G., Bischof W.F., Yang X.D.: Hapto-audio-visual environments for collaborative training of ophthalmic surgery over optical network in: Haptic Audio Visual Environments and their Applications, 2006. HAVE 2006. IEEE International Workshop on (2006), 21–26
5. Bowman D., Kruijff E., LaViola J., Poupyrev I.: 3D User Interfaces: Theory and Practice. Addison-Wesley (2005), ISBN 0-201-75867-9.
6. Christoffersen K., Woods D.D.: How to make automated systems team players, Advances in human performance and cognitive engineering research (2002), 1–12

7. Cobb S., D'Cruz M., Day A., David P., Gardeux F., van de Broek E. L., van de Voort M. C., Meijer F., Izkara J. L., Mavrikios D.: How is vr used to support training in industry?, The intuition network of excellence working group on education and training in VRIC'08 (2008), 75–83
8. Donikian S.: HPTS: a behaviour modelling language for autonomous agents Proceedings of the fifth international conference on Autonomous agents (2001) 401–408
9. Donikian S.: Modelling, control and animation of autonomous virtual agents evolving in structured and informed environments Habilitation to supervise research in computer science, University of Rennes 1 (2004)
10. Dugdale J., Pavard B., Pallamin N., El Jed M.: Emergency fire incident training in a virtual world in: Proceedings ISCRAM2004 (2004)
11. Dugdale J., Darcy S., Pavard B.: Engineering effective cooperation and communication: a bottom-up approach (2006), 58–65
12. Dugdale J., Bellamine-Ben Saoud N., Pavard B., Pallamin N.: Simulation and Emergency Management (2009), 229
13. Duval T., Le Tenier C.: 3D Collaborative interactions within virtual environments with OpenMASK for technical objects exploitation Mécanique & Industries (2004), 767-797
14. Gerbaud S., Mollet N., Arnaldi B.: Virtual Environments for Training: From Individual Learning to Collaboration with Humanoids in: Edutainment (2007),
15. Gerbaud S., Gouranton V., Arnaldi B.: Adaptation in Collaborative Virtual Environments for Training in: Edutainment (2009), 316-327
16. Granieri J. P., Becket W., Reich B. D., Crabtree J., Badler N. I.: Behavioral control for real-time simulated human agents. In SI3D '95 (1995), ACM Press, pp. 173–180.
17. Kallmann M., Thalmann D.: Modeling Objects for Interaction Tasks Proc. Eurographics Workshop on Animation and Simulation (1998) 73–86
18. Margery D., Arnaldi B., Plouzeau N.: A general framework for cooperative manipulation in virtual environments. Virtual Environments 99 (1999) 169–178
19. Mollet N., Gerbaud S., Arnaldi B.: STORM: a Generic Interaction and Behavioral Model for 3D Objects and Humanoids in a Virtual Environment IPT-EGVE the 13th Eurographics Symposium on Virtual Environments (2007) 95–100
20. Mollet N., Gerbaud S., Arnaldi B.: An operational vr platform for building virtual training environments in: Proceedings of the 2nd international conference on Technologies for e-learning and digital entertainment (2007), 140–151
21. Oliveira J.C., Hosseini S., Shirmohammadi M., Cordea M., Petriu N.D., Petriu E., Georganas D.C.: Virtual theater for industrial training: A collaborative virtual environment (2000)
22. Pinho M.S., Bowman D.A., Freitas C.M.D.S.: Cooperative Object Manipulation in Immersive Virtual Environments in: Virtual Reality Software and Technology Proceedings of the ACM symposium on Virtual reality software and technology Hong Kong, China (2002), 171–178
23. Querrec R., Buche C., Maffre E., Chevaillier P.: SécuRévi : virtual environments for fire-fighting training in: Proceedings of the 5th Virtual Reality International Conference, VRIC 2003 (2003), Computer Graphics Forum (1998), 41–52
24. Reynolds C. W.: Flocks, herds, and schools: A distributed behavioral model. Computer Graphics 21, 4(1987), 25–34
25. Sales B.R.A., Machado L.S., Moraes R.M.: Interactive collaboration for virtual reality systems related to medical education and training in: 6th Int. Conf. on Tech. and Medical Sciences. Porto (2010)
26. Sycara K., Sukthankar G.: Literature Review of Teamwork Models Robotics Institute (2006)